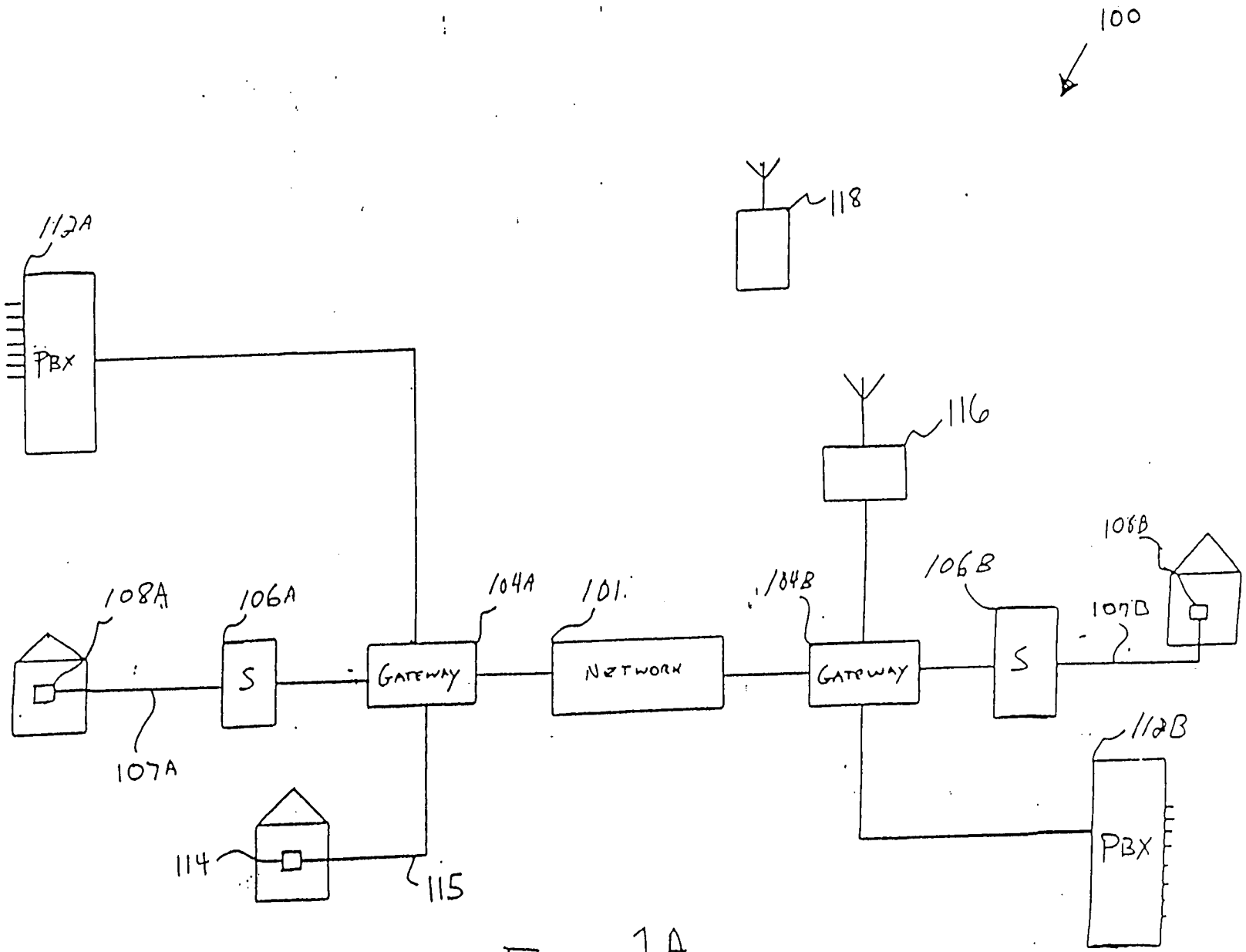
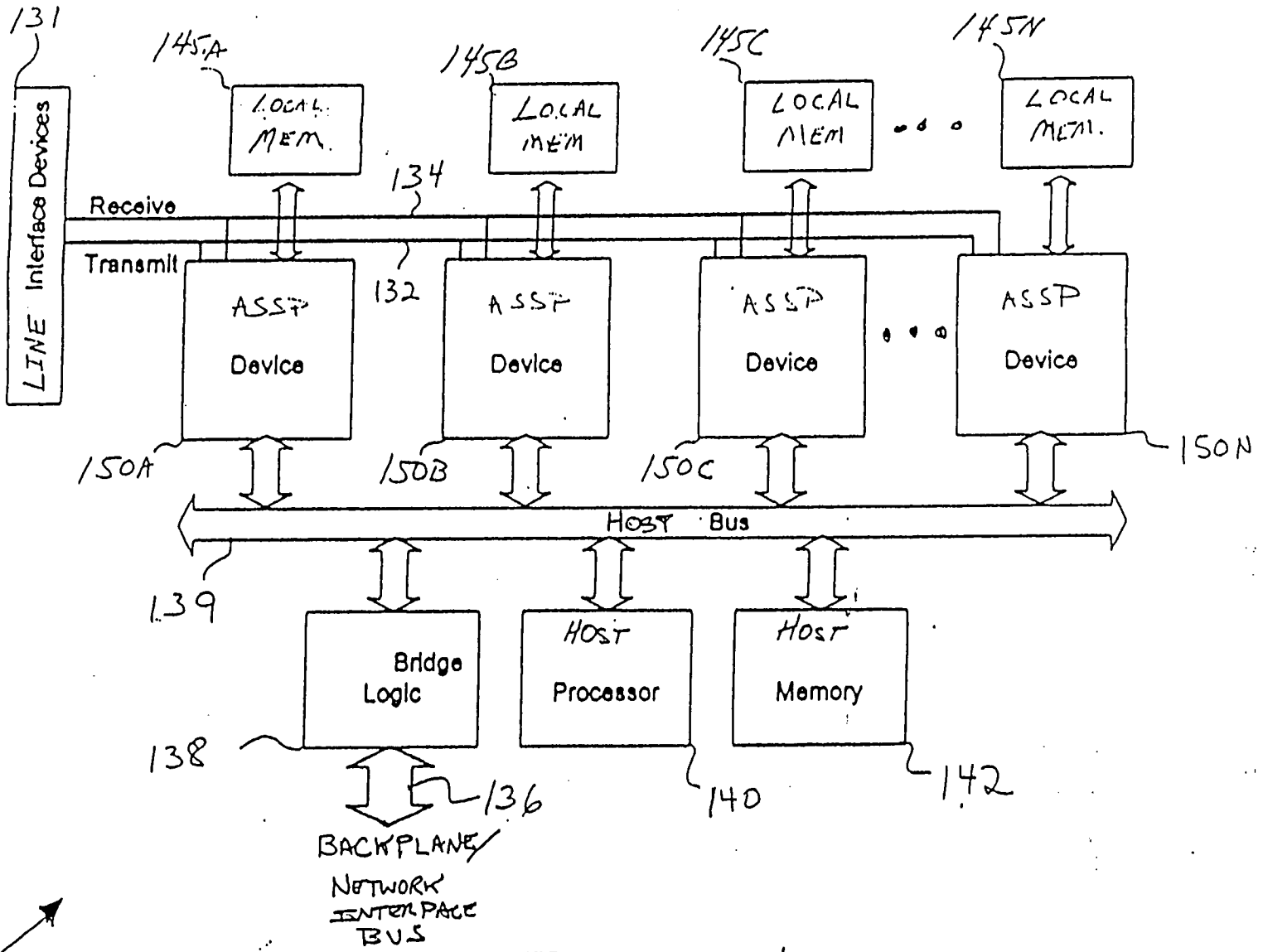

100
118
112A
PBX
116
108B
106B
108A
106A
104A
101
104B
107B
S
GATEWAY
NETWORK
GATEWAY
S
107A
112B
114
115
PBX

FIG. 1A


131
LINE Interface Devices
145A
LOCAL MEM.
145B
LOCAL MEM
145C
LOCAL MEM
145N
LOCAL MEM.
134
Receive
Transmit
132
ASSP Device
ASSP Device
ASSP Device
ASSP Device
150A
150B
150C
150N
Host Bus
139
Bridge Logic
Host Processor
Host Memory
138
136
140
142
BACKPLANE
NETWORK INTERFACE BUS

130

FIG. 1B

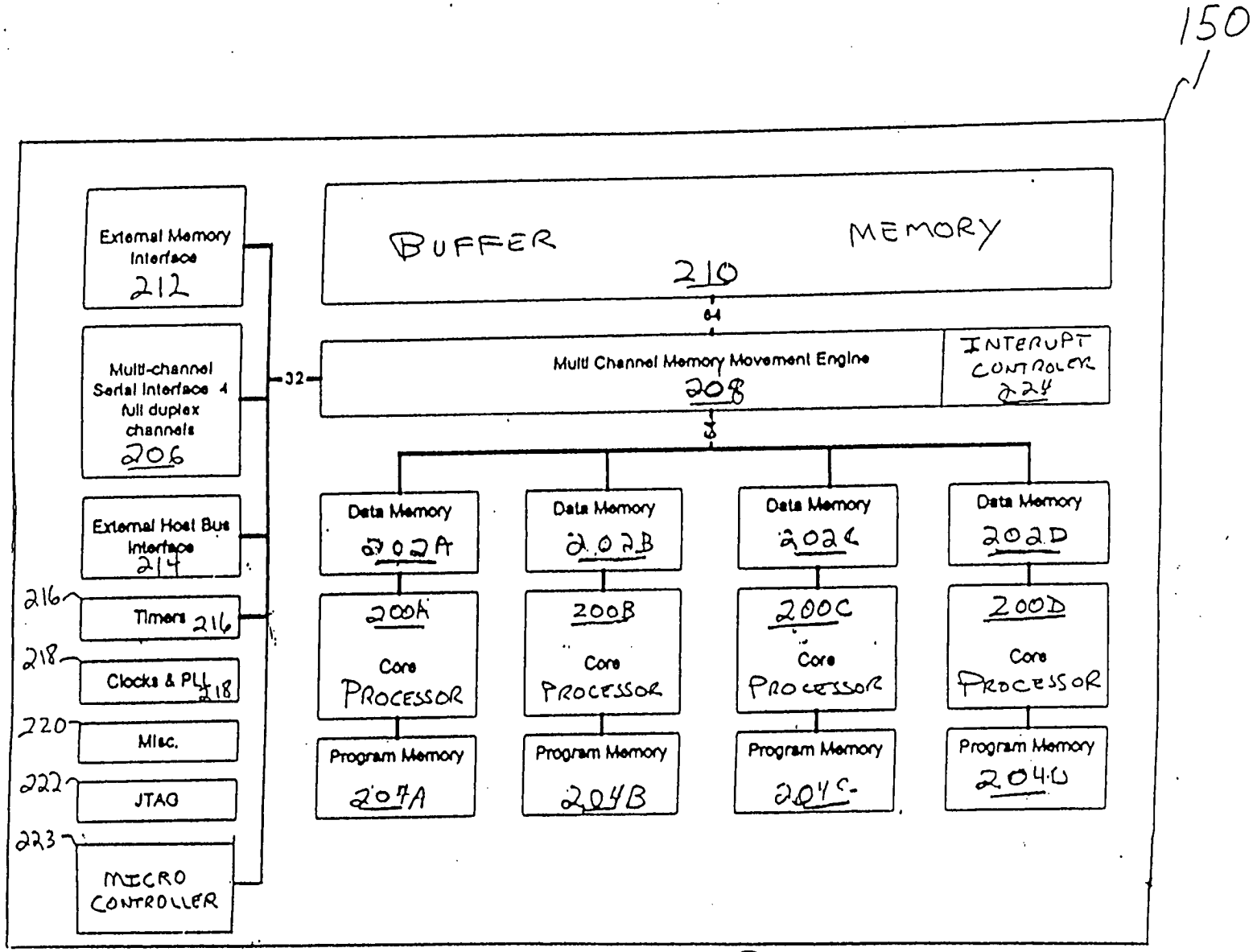

150
External Memory Interface 212
Multi-channel Serial Interface 4 full duplex channels 206
External Host Bus Interface 214
216
Timers 216
218
Clocks & PLL 218
220
Misc.
222
JTAG
223
MICRO CONTROLLER
32
BUFFER MEMORY 210
64
Multi Channel Memory Movement Engine 208
INTERUPT CONTROLER 224
8
Data Memory 202A
Data Memory 202B
Data Memory 202C
Data Memory 202D
200A Core PROCESSOR
200B Core PROCESSOR
200C Core PROCESSOR
200D Core PROCESSOR
Program Memory 204A
Program Memory 204B
Program Memory 204C
Program Memory 204D

FIG. 2

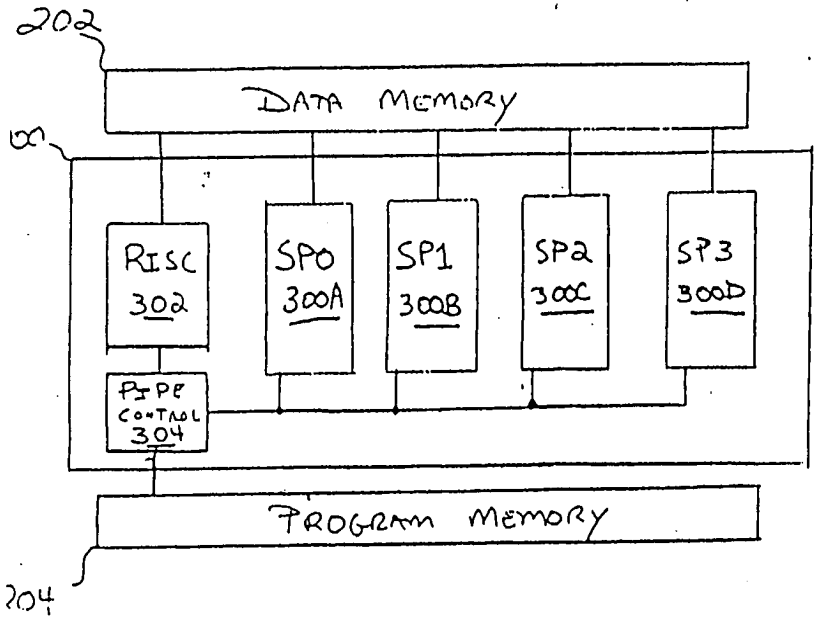


FIG. 3

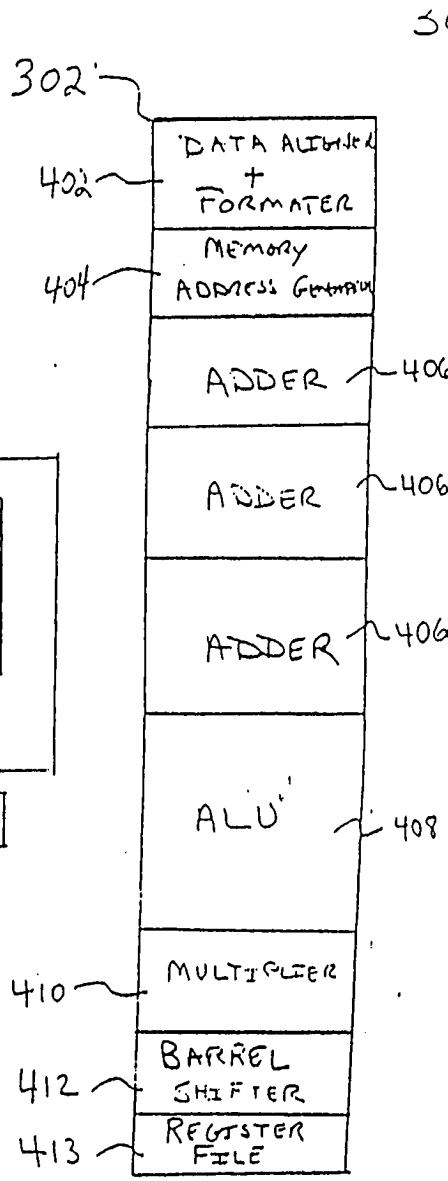


FIG. 4

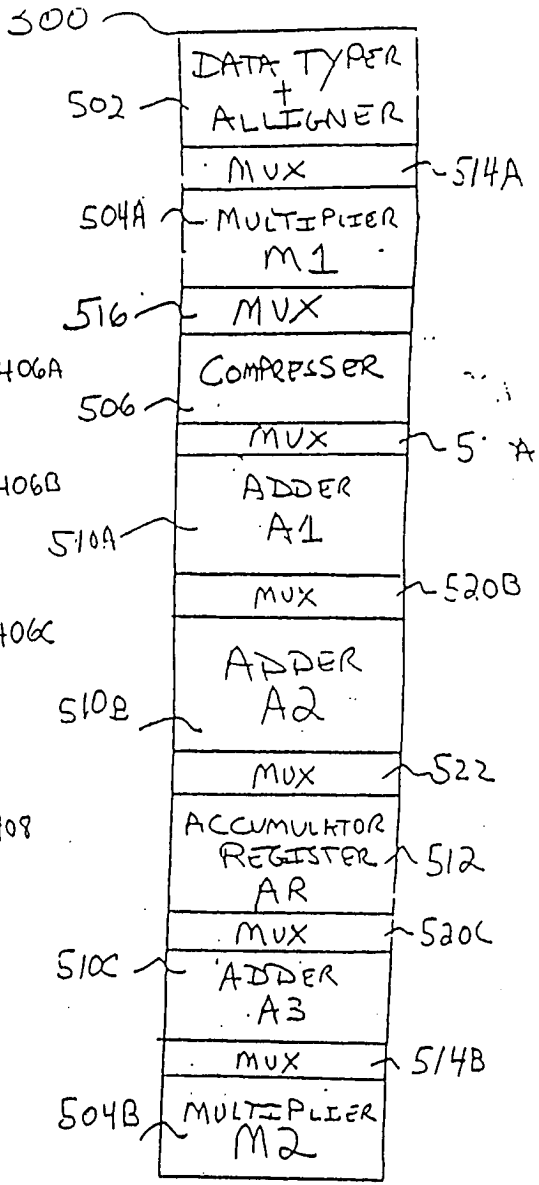


FIG. 5A

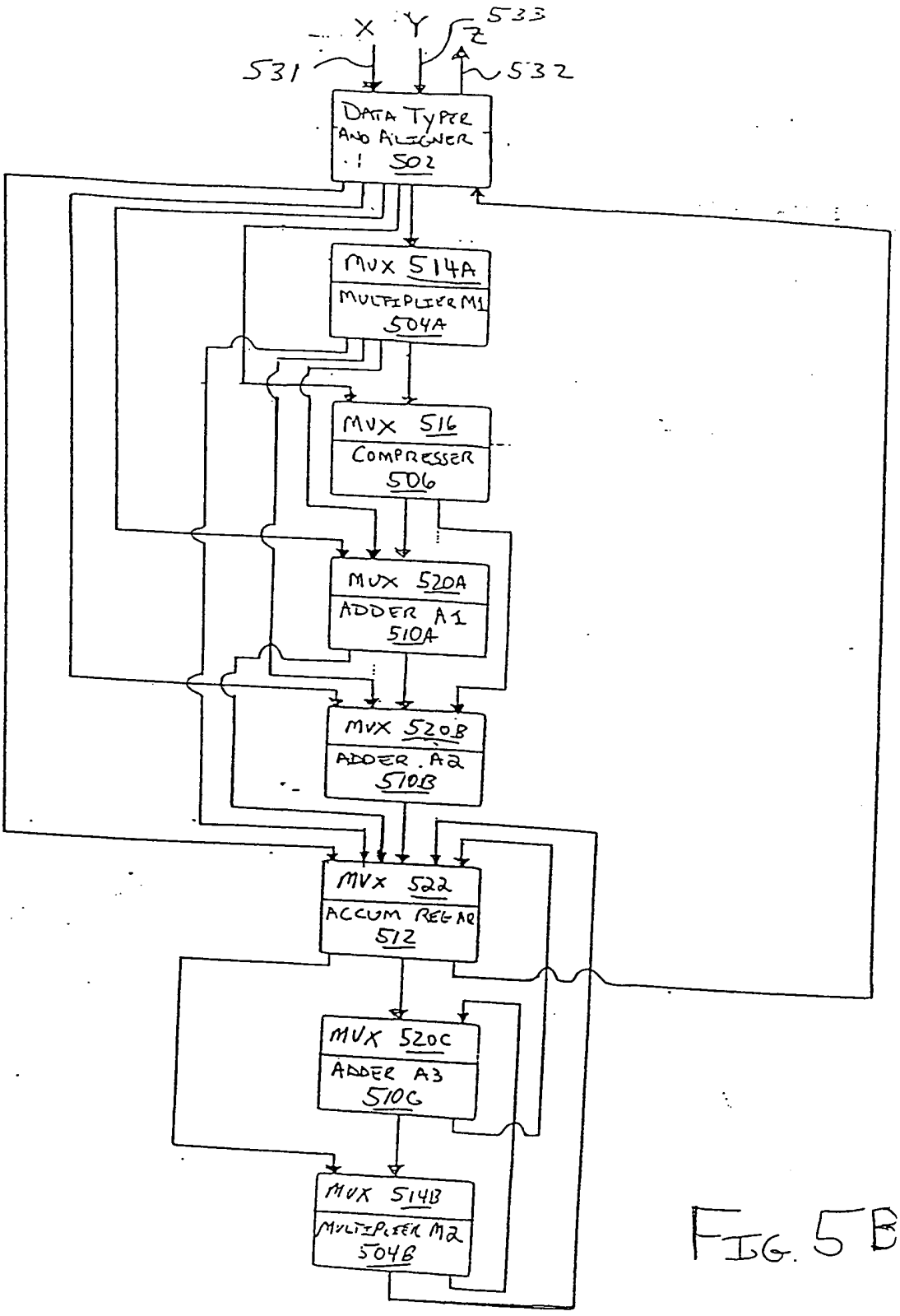

X
Y
Z
533
531
532
DATA TYPE AND ALIGNER 502
MUX 514A
MULTIPLIER M1 504A
MUX 516
COMPRESSER 506
MUX 520A
ADDER A1 510A
MUX 520B
ADDER A2 510B
MUX 522
ACCUM REGAR 512
MUX 520C
ADDER A3 510C
MUX 514B
MULTIPLIER M2 504B

FIG. 5B

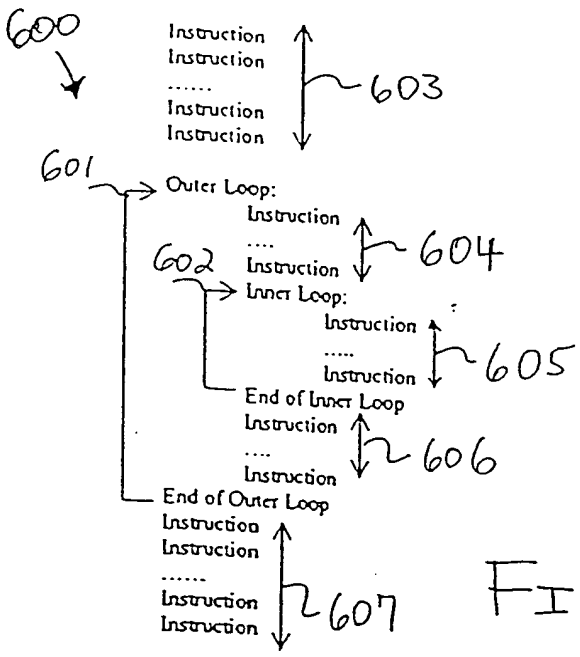


FIG. 6A

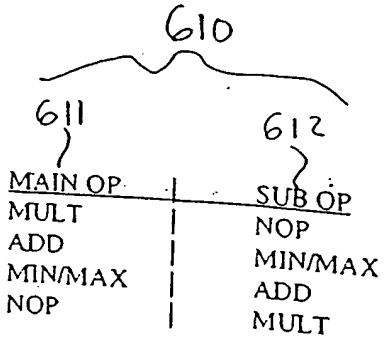


FIG. 6B

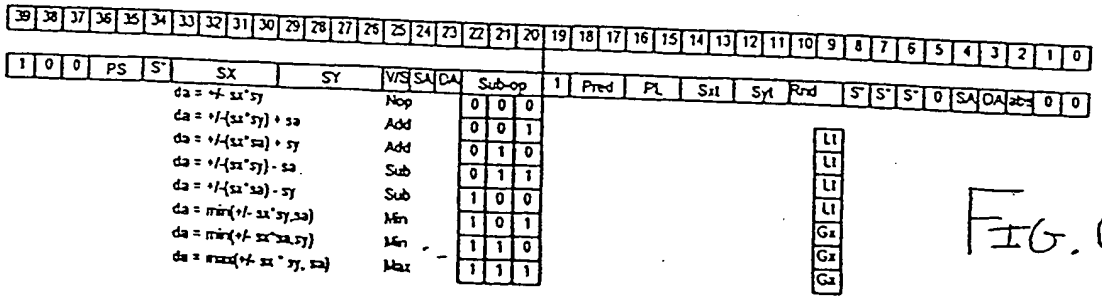


FIG. 6C

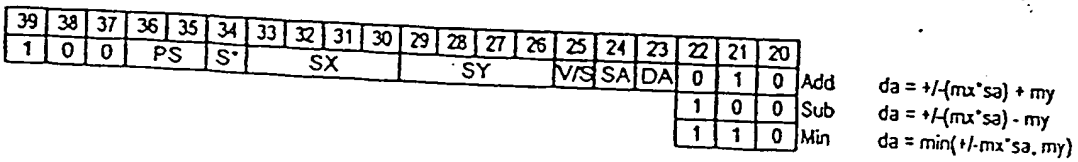
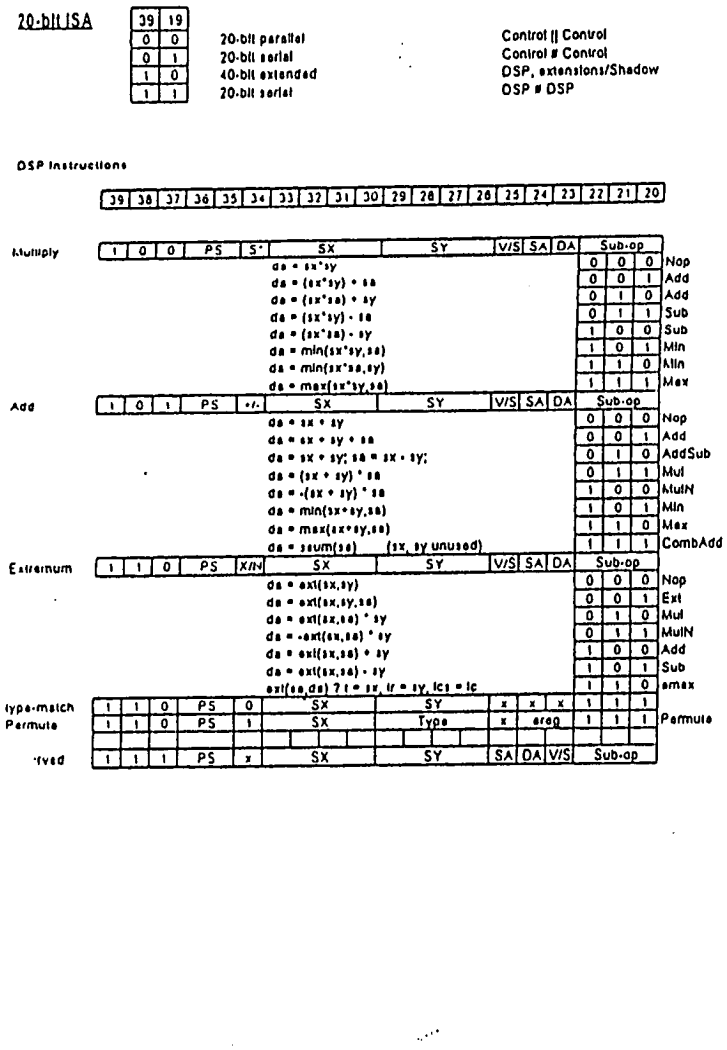


FIG. 6D


20-bit ISA
39 19
0 0 20-bit parallel
0 1 20-bit serial
1 0 40-bit extended
1 1 20-bit serial
Control || Control
Control # Control
DSP, extensions/Shadow
OSP # DSP
DSP Instructions
39 38 37 36 35 34 33 32 31 30 29 28 27 26 25 24 23 22 21 20
Multiply
1 0 0 PS S* SX SY V/S SA DA Sub-op
da = sx*sy Nop
da = (sx*sy) + sa Add
da = (sx*sa) + sy Add
da = (sx*sy) - sa Sub
da = (sx*sa) - sy Sub
da = min(sx*sy,sa) Min
da = min(sx*sa,sy) Min
da = max(sx*sy,sa) Max
Add
1 0 1 PS +/- SX SY V/S SA DA Sub-op
da = sx + sy Nop
da = sx + sy + sa Add
da = sx + sy; sa = sx + sy; AddSub
da = (sx + sy) * sa Mul
da = -(sx + sy) * sa MulN
da = min(sx+sy,sa) Min
da = max(sx+sy,sa) Max
da = sum(sa) (sx, sy unused) CombAdd
Extremum
1 1 0 PS X/N SX SY V/S SA DA Sub-op
da = ext(sx,sy) Nop
da = ext(sx,sy,sa) Ext
da = ext(sx,sa) * sy Mul
da = -ext(sx,sa) * sy MulN
da = ext(sx,sa) + sy Add
da = ext(sx,sa) - sy Sub
ext(sa,da) ? t = sx, lr = sy, lcs = lc emax
type-match
Permute
1 1 0 PS 0 SX SY x x x 1 1 1
1 1 0 PS 1 SX Type x areg 1 1 1 Permute
rved
1 1 1 PS x SX SY SA DA V/S Sub-op

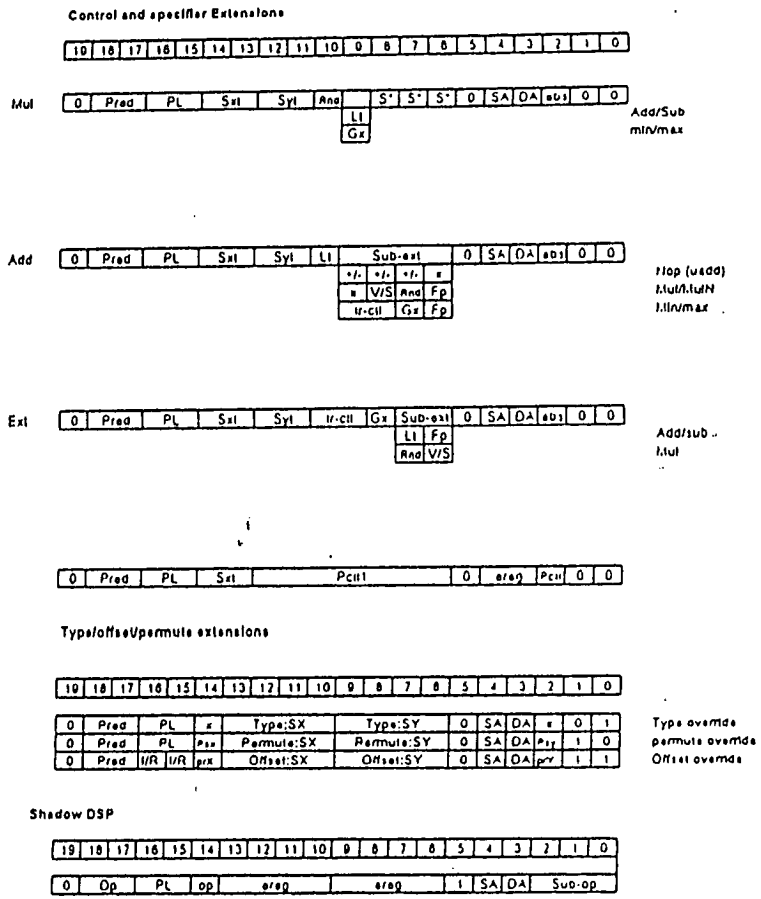

Control and specifier Extensions
10 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Mul
0 Pred PL Sx1 Sy1 Rnd LI Gx S* S* S* 0 SA DA adj 0 0
Add/Sub
min/max
Add
0 Pred PL Sx1 Sy1 LI Sub-ext 0 SA DA adj 0 0
+/- +/- +/- x
x V/S Rnd Fp
lr-clr Gx Fp
flop (uadd)
Mul/MulN
Min/max
Ext
0 Pred PL Sx1 Sy1 lr-clr Gx Sub-ext 0 SA DA adj 0 0
LI Fp
Rnd V/S
Add/sub
Mul
0 Pred PL Sx1 Pclr 0 areg Pclr 0 0
Type/offset/permute extensions
10 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
0 Pred PL x Type:SX Type:SY 0 SA DA x 0 1
Type override
0 Pred PL Psx Permute:SX Permute:SY 0 SA DA Psy 1 0
permute override
0 Pred lr/R lr/R prx Offset:SX Offset:SY 0 SA DA pry 1 1
Offset override
Shadow DSP
19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
0 Op PL op areg areg 1 SA DA Sub-op

FIG. 6 E

Control Instructions

	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
add,sub	L	Pred		0	0	0	RX				RY				RZ				+/-	0
max,min	L	Pred		0	0	0	RX				RY				RZ				X/N	1
Shift	L	Pred		0	0	1	RX				UI4				RZ				UI1	R/L
Logic	L	Pred		0	1	0	RX				RY				RZ				&, |	&|
Mux	L	Pred		0	1	1	RX				RY				RZ				Pa	0
mov	L	Pred		0	1	1	RX				OZ				Rxi	Dzi	0	0	0	1
addi	L	Pred		0	1	1	SI4				OZ				x	x	1	0	0	1
mov2erg	L	Pred		0	1	1	RX				unit		ereg		gd	type		1	0	1
Ldm	L	Pred		0	1	1	RX				OZ1				OZ2				1	1
tbits	L	Pred		1	0	0	UI4:POS				RZ				Rzi	UI4				0
bits	L	Pred		1	0	0	UI4:POS				RZ				Rzi	UI2		0	0	1
Setbit	L	Pred		1	0	0	UI4:POS				RZ				Rzi	UI1	UI1	1	0	1
Movl	L	Pred		1	0	0	SI8								RZ				-1	1
Jmp	L	Pred		1	0	1	SI8								0	Pred			0	0
Call	L	Pred		1	0	1	SI8								1	Pred			0	0
Loop	L	Pred		1	0	1	UI5 : Lcount				UI5: Lsize							UI2; Lsl	0	1
Jmpi	L	Pred		1	0	1	RX			x	x	x	x	x	0	Pred			1	0
Calli	L	Pred		1	0	1	RX			x	x	x	x	x	1	Pred			1	0
Loopi	L	Pred		1	0	1	RX			x	UI5; Lsize							UI2: Lsl	1	1
Test	L	Pred		1	1	0	RX				RY				PZ			= < >		0
Testbit	L	Pred		1	1	0	RX				UI5				PZ			B	0	1
Andp, orp	L	Pred		1	1	0	Pa			Pb			Pc		PZ			&|	1	1
Load	L	Pred		1	1	1	MX				RZ				Ext			0	0	0
Store	L	Pred		1	1	1	MZ				RX				Ext			1	0	0
eLoad	L	Pred		1	1	1	MX				RZ				1	1	1	0	0	0
eStore	L	Pred		1	1	1	MZ				RX				1	1	1	1	0	0
Extended	L	Pred		1	1	1	Bits 27:16												1	0
Logic2	L	Pred		1	1	1	RX				RY/RZ				Rxi	Ryi	&, |, &|		0	1
mov-erg	L	Pred		1	1	1	unit		ereg		RZ				gd	Sft		0	1	1
Crc	L	Pred		1	1	1	RX				RZ				s/m	0	0	1	1	1
Parity	L	Pred		1	1	1	RX				PZ			O/E	0	1	0	1	1	1
Sim	L	Pred		1	1	1	MZ				RX				1	1	0	1	1	1
Abs	L	Pred		1	1	1	RX				RZ				0	0	1	1	1	1
Neg	L	Pred		1	1	1	RX				RZ				0	1	1	1	1	1
Div-step	L	Pred		1	1	1	RX				RZ				1	0	1	1	1	1
& Sel	L	Pred		1	1	1	RX				PZ			0	1	1	1	1	1	1
[illegible]	L	Pred		1	1	1	RX				RZ			1	1	1	1	1	1	1
Return	L	Pred		1	1	1	Pred		I-ctl		0	1	0	1	1	1	1	1	1	1
Zero-ac	L	Pred		1	1	1	ac #				1	1	0	1	1	1	1	1	1	1
eSync	L	Pred		1	1	1	RZ				0	1	1	1	1	1	1	1	1	1
Swi	L	Pred		1	1	1	UI3			0	1	1	1	1	1	1	1	1	1	1
Nop	L	Pred		1	1	1	UI3			1	1	1	1	1	1	1	1	1	1	1

<Bit1, Bits9-6> == UI5 (Shift Amount)

<Bit3, Bits13-10> == UI5 :POS

FIG. 6F

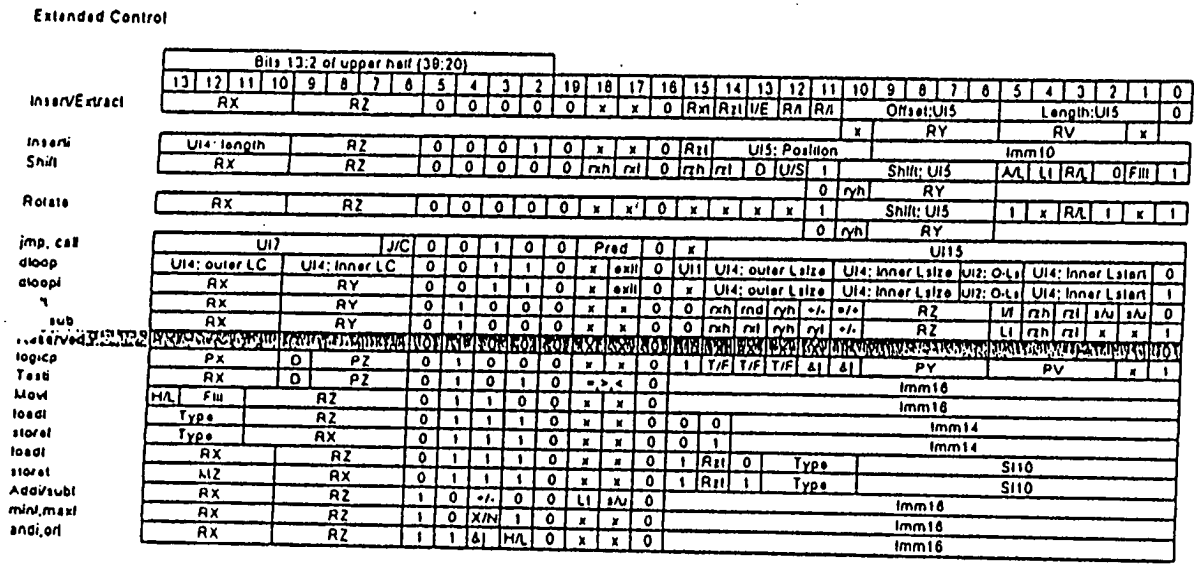

Extended Control
Bits 13:2 of upper half (39:20)
InserVExtract
Inseri
Shift
Rotate
jmp, call
dloop
aloopi
sub
logicp
Testi
Movi
loadi
storei
loadi
storei
Addisubi
mini,maxi
andi,ori
Fill: Sign/Zero
Bit 15 is continuation of inner LC
andp, orp, andorp, orandp: pz = (px relop py) relop pv)

FIG. 6G

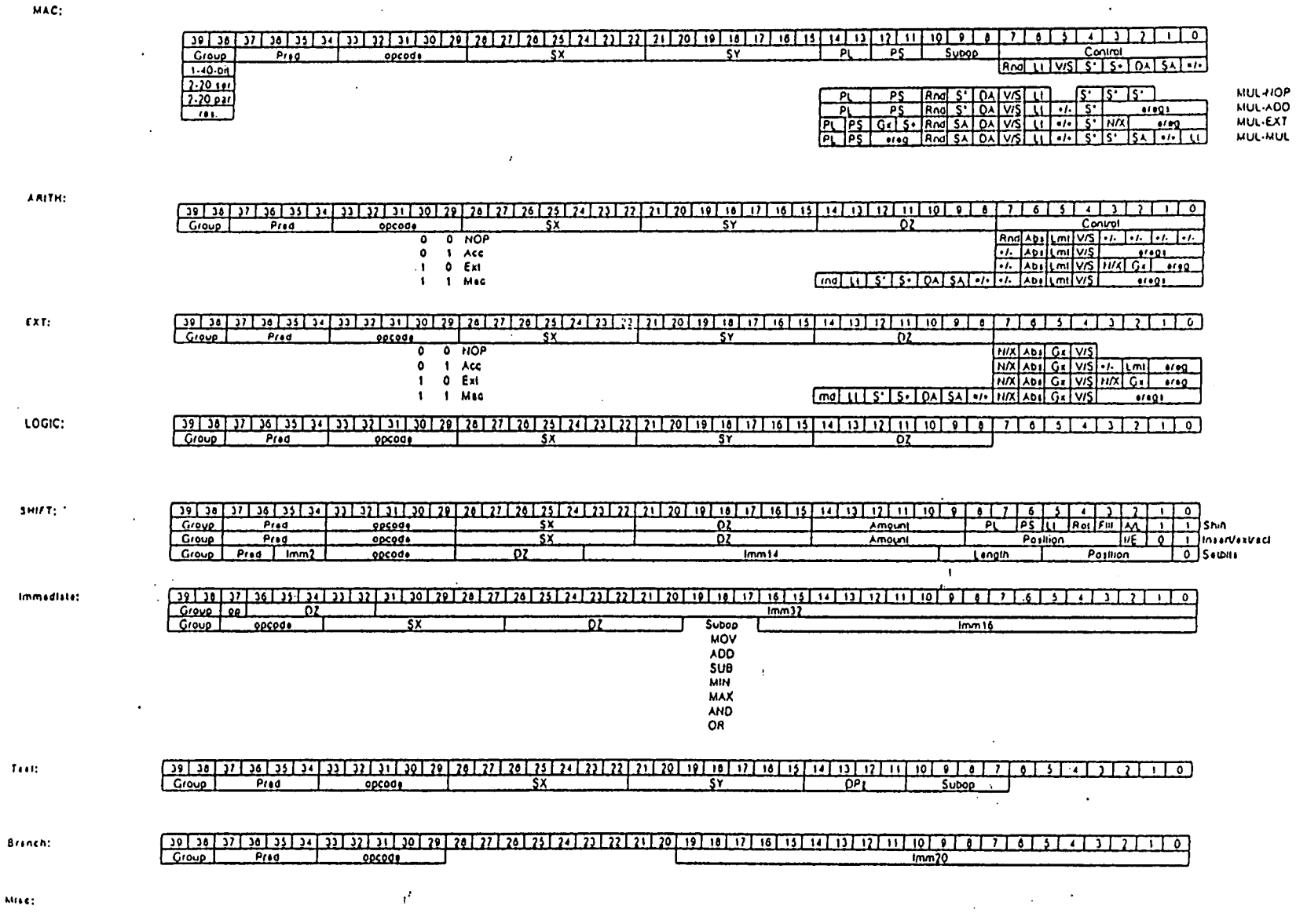

MAC:
ARITH:
EXT:
LOGIC:
SHIFT:
Immediate:
Test:
Branch:
Misc:
MUL-NOP
MUL-ADD
MUL-EXT
MUL-MUL
NOP
Acc
Ext
Mac
Shift
Insert/Extract
Setbits
MOV
ADD
SUB
MIN
MAX
AND
OR

FIG. 6H

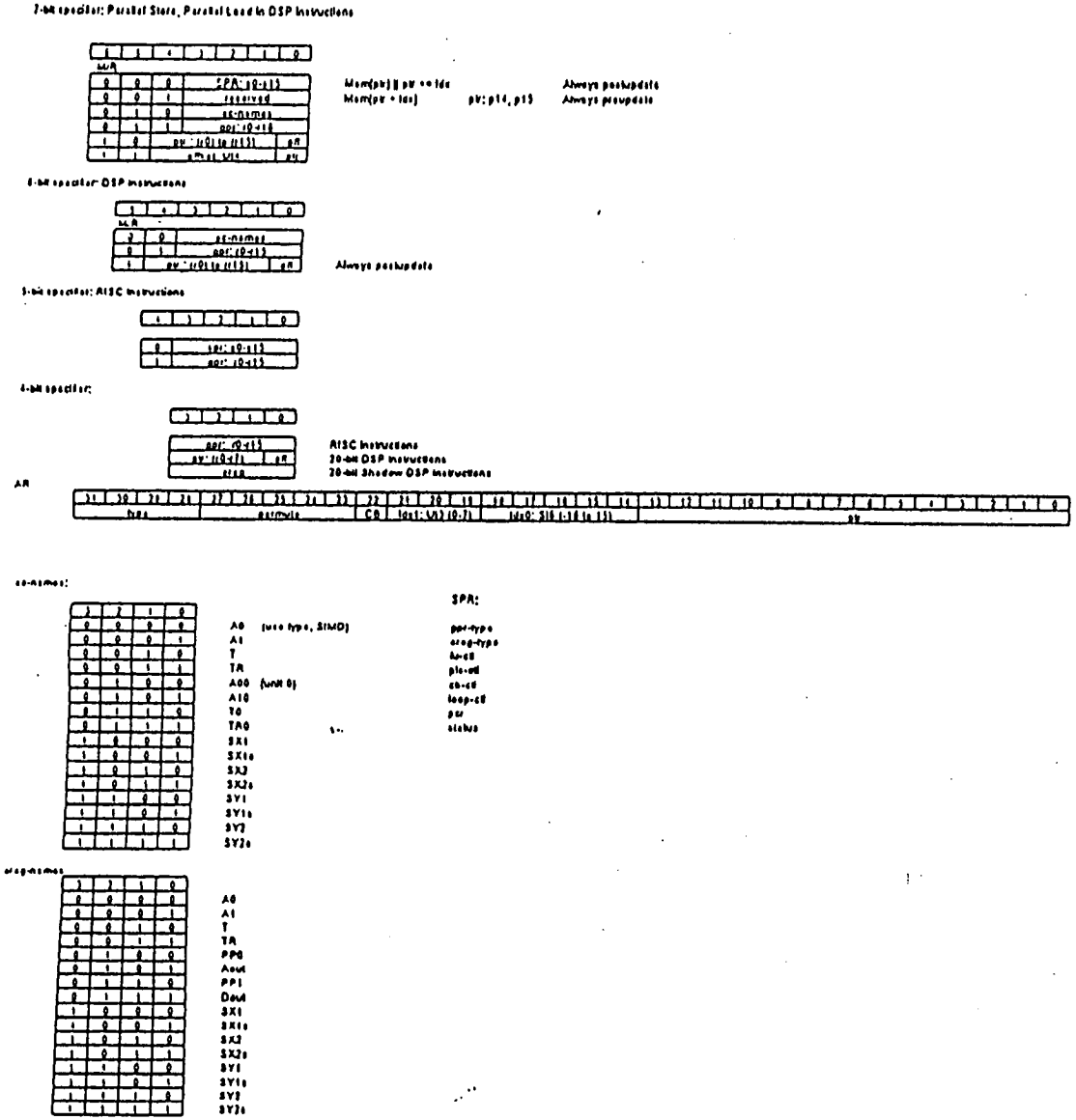


FIG. 6I

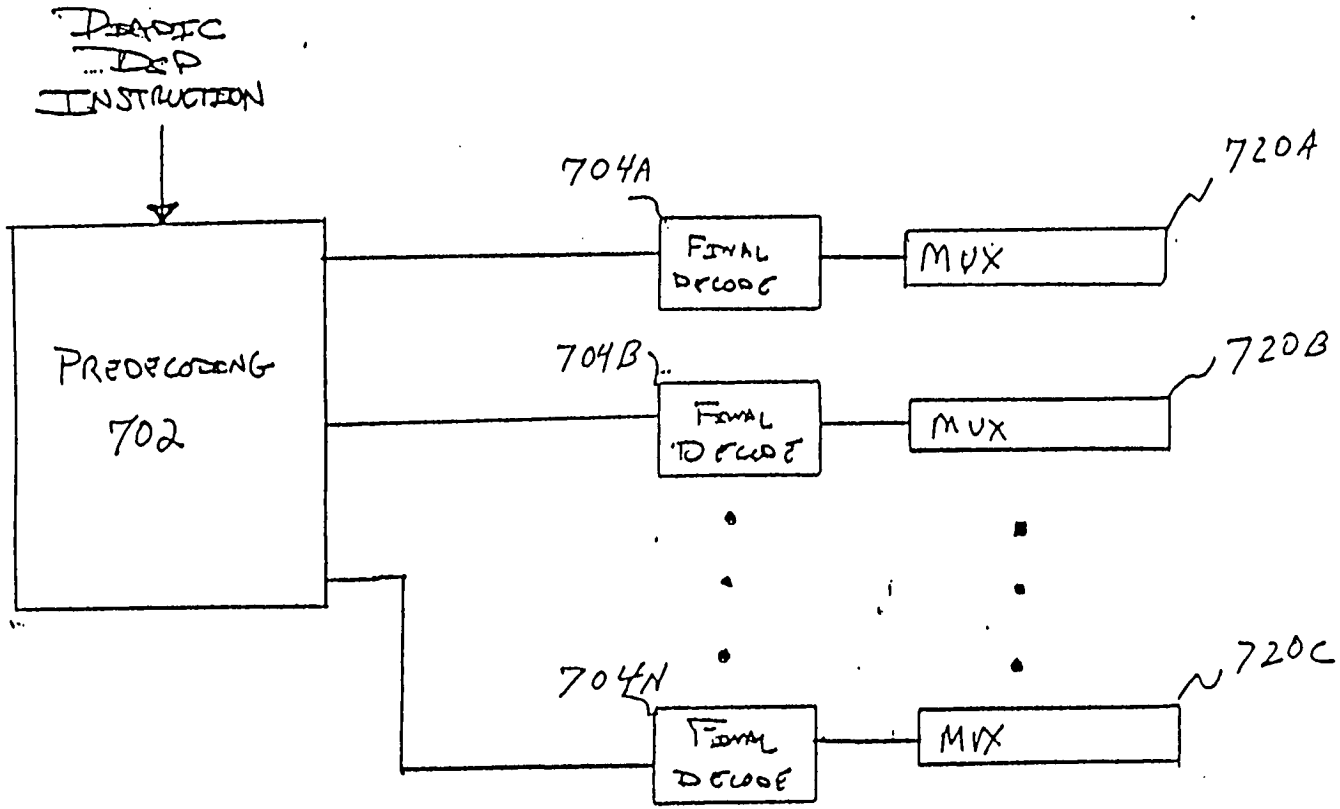

DSP
INSTRUCTION
PREDECODING
702
704A
FINAL
DECODE
720A
MUX
704B
FINAL
DECODE
720B
MUX
704N
FINAL
DECODE
720C
MUX

FIG. 7

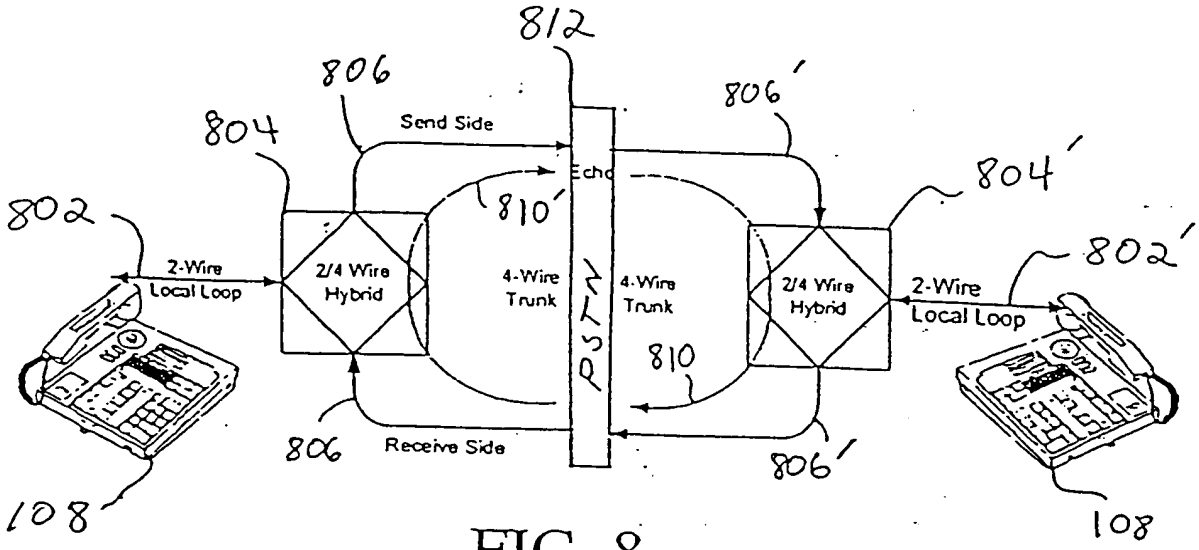


FIG. 8
(PRIOR ART)

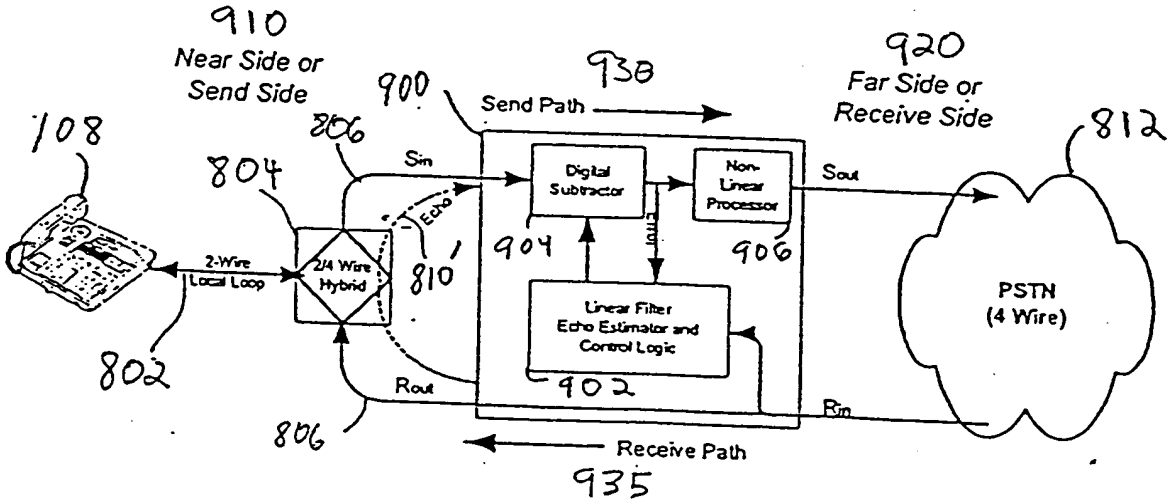


FIG. 9
(PRIOR ART)

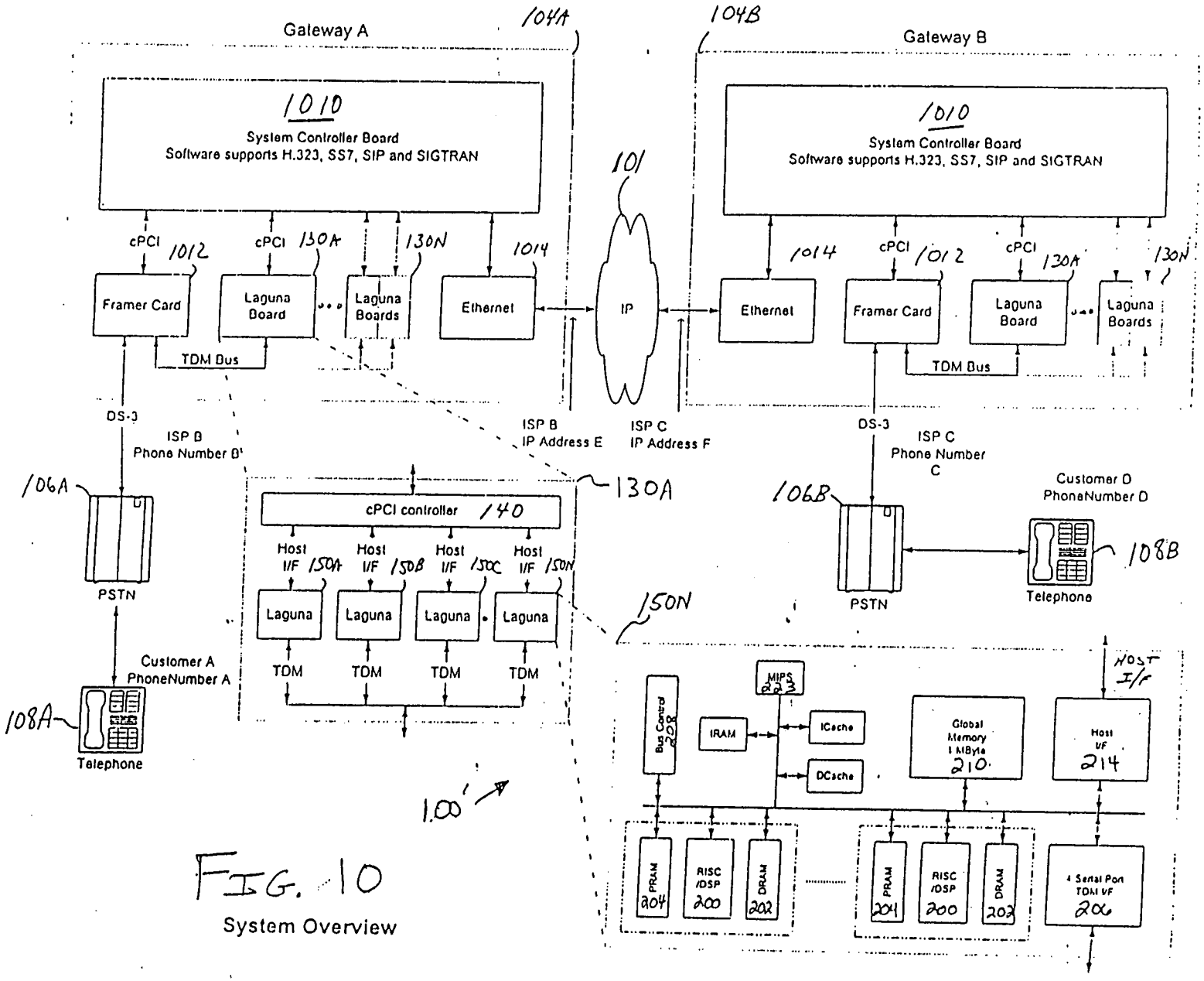


FIG. 10

System Overview

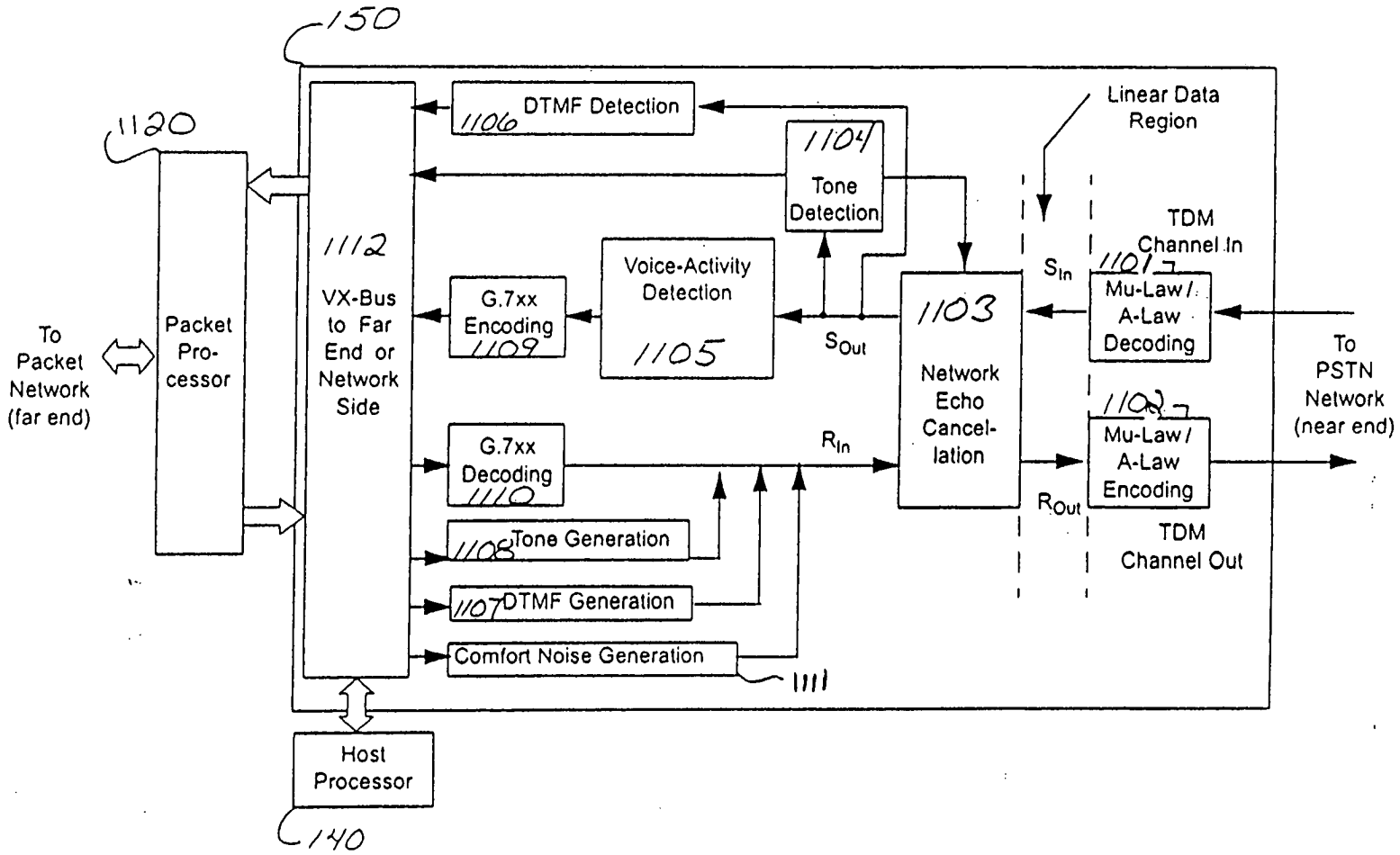

150
1120
1112
VX-Bus to Far End or Network Side
Packet Pro-cessor
To Packet Network (far end)
DTMF Detection
1106
1104
Tone Detection
Linear Data Region
TDM Channel In
1101
Mu-Law / A-Law Decoding
To PSTN Network (near end)
G.7xx Encoding
1109
Voice-Activity Detection
1105
SOut
SIn
1103
Network Echo Cancel-lation
1102
Mu-Law / A-Law Encoding
TDM Channel Out
G.7xx Decoding
1110
RIn
ROut
Tone Generation
1108
DTMF Generation
1107
Comfort Noise Generation
1111
Host Processor
140

FIG. 11A

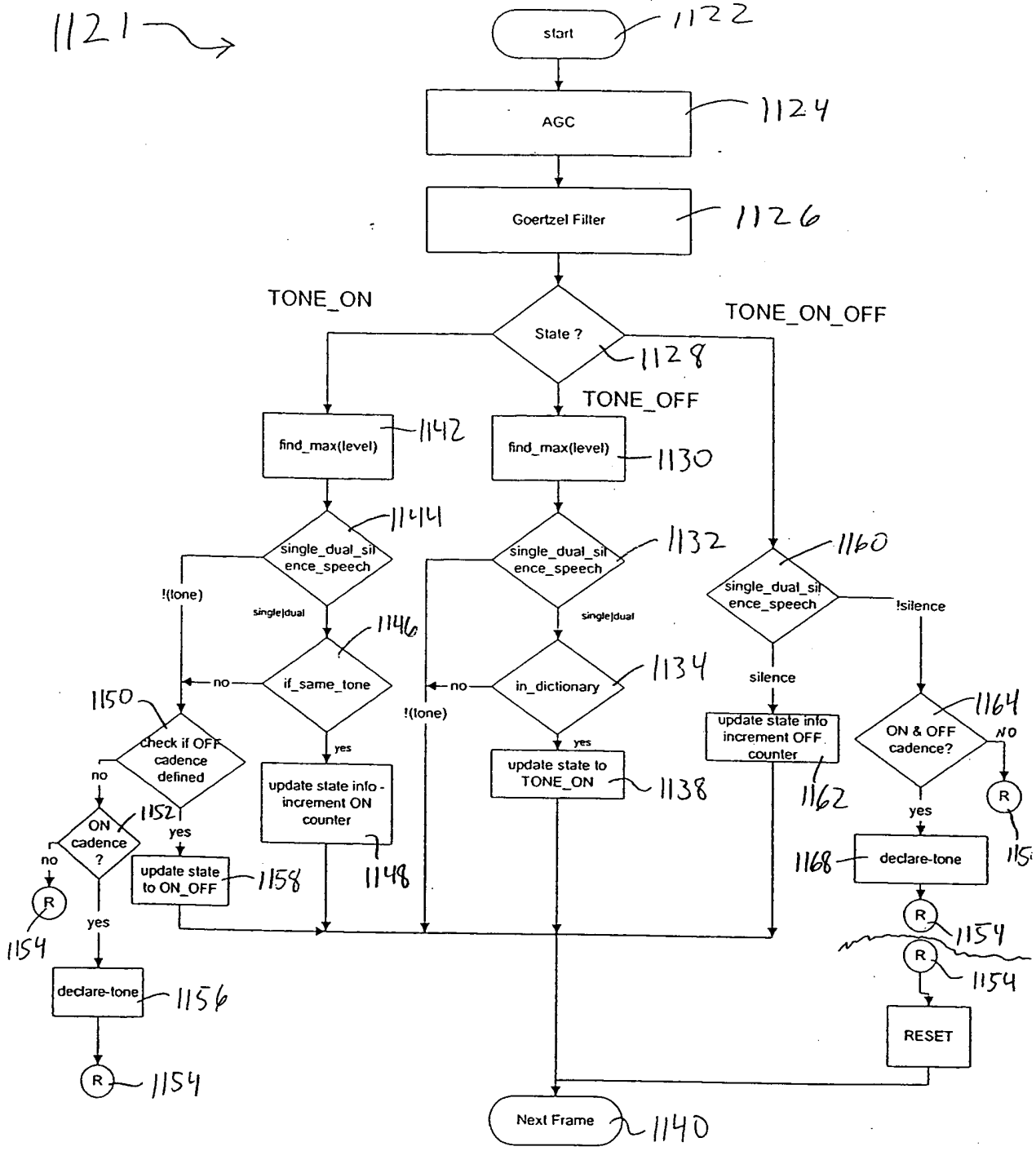

1121
start
1122
AGC
1124
Goertzel Filter
1126
TONE_ON
State ?
1128
TONE_ON_OFF
TONE_OFF
find_max(level)
1142
find_max(level)
1130
single_dual_silence_speech
1144
single_dual_silence_speech
1132
single_dual_silence_speech
1160
!(tone)
single|dual
no
if_same_tone
1146
yes
update state info - increment ON counter
1148
no
in_dictionary
1134
yes
update state to TONE_ON
1138
silence
update state info increment OFF counter
1162
!silence
ON & OFF cadence?
1164
NO
R
1165
yes
declare-tone
1168
R
1154
R
1154
RESET
1150
check if OFF cadence defined
no
ON cadence ?
1152
yes
update state to ON_OFF
1158
no
R
1154
yes
declare-tone
1156
R
1154
Next Frame
1140

FIG. 11B

Exemplary Filter coefficients for Goertzel Filter

frequency	cos(2*pi*f1/fs)	frequency index
350	31536	0
400	31163	1
425	30958	2
440	30829	3
480	30465	4
540	29863	5
600	29195	6
620	28958	7
660	28462	8
697	27978	9
700	27938	10
770	26955	11
780	26808	12
852	25700	13
900	24916	14
941	24218	15
1020	22802	16
1100	21280	17
1140	20487	18
1209	19072	19
1300	17120	20
1336	16324	21
1380	15332	22
1477	13084	23
1500	12539	24
1620	9634	25
1633	9314	26
1700	7649	27
1740	6644	28
1860	3595	29
1980	514	30
2040	-1029	31
2100	-2570	32
2280	-7147	33
2400	-10125	34
2600	-14875	35
3825	-32457	36

FIG. 11C

Exemplary Call Progress Tones

Frequency1	Frequency2	Call Progress Tone
350	440	ANSI T1.401 dial tone
425	0	Q.35 Dial Tone
440	480	ANSI T1.401 audible ringing
480	620	ANSI T1.401line busy tone
480	620	ANSI T1.401Reorder
400	0	Audible ringing
440	0	Dial Tone
440	0	ANSI T1.401Fast Busy Tone
440	0	Busy Tone

FIG. 11D

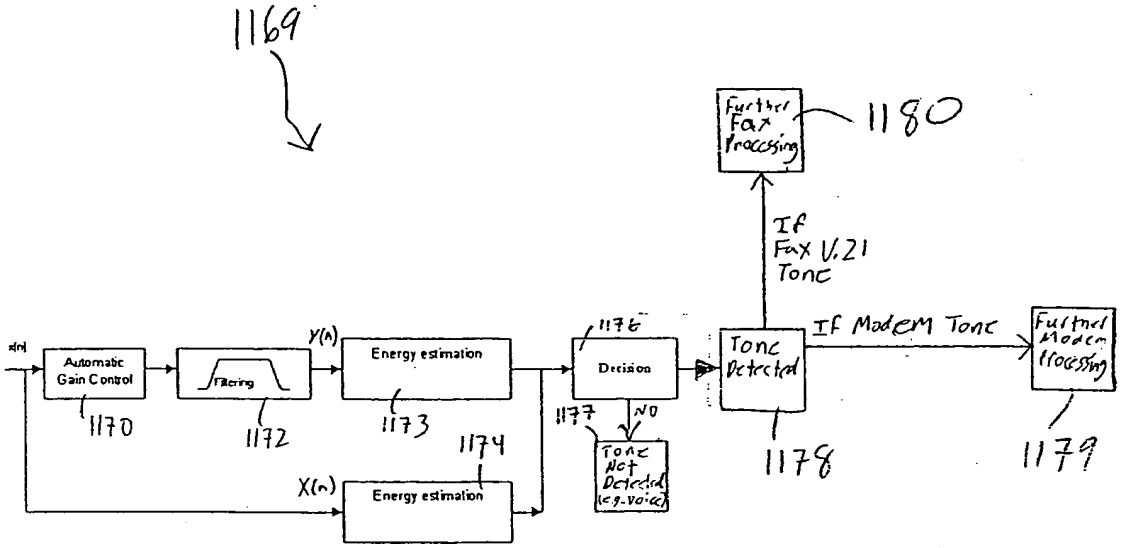

1169
Further Fax Processing
1180
If Fax V.21 Tone
If Modem Tone
Further Modem Processing
x[n]
Automatic Gain Control
1170
Filtering
1172
Y(n)
Energy estimation
1173
1176
Decision
1177
NO
Tone Not Detected (e.g. voice)
1174
X(n)
Energy estimation
Tone Detected
1178
1179

FIG. 11E

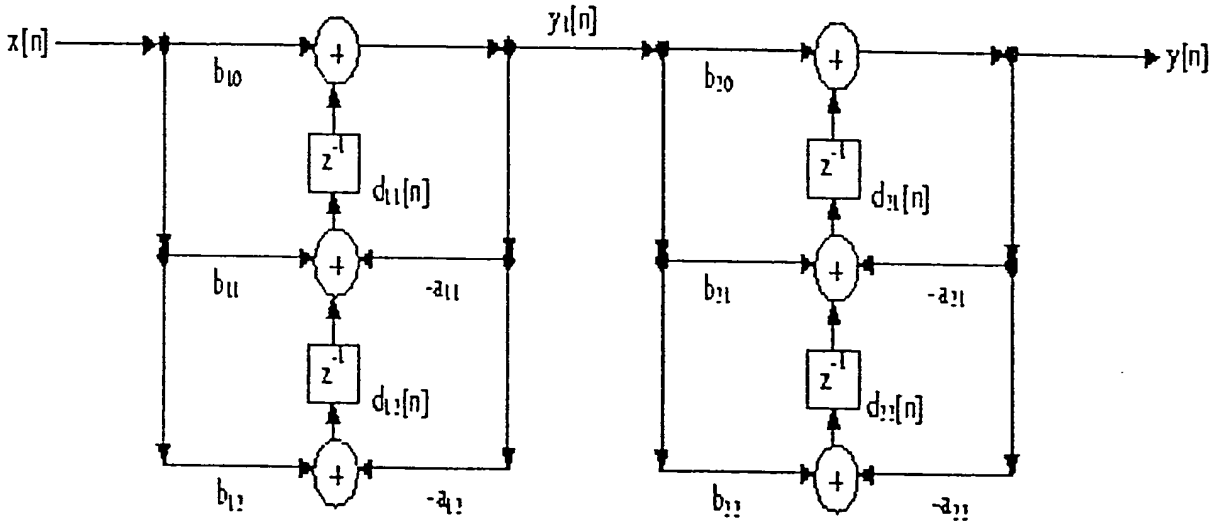
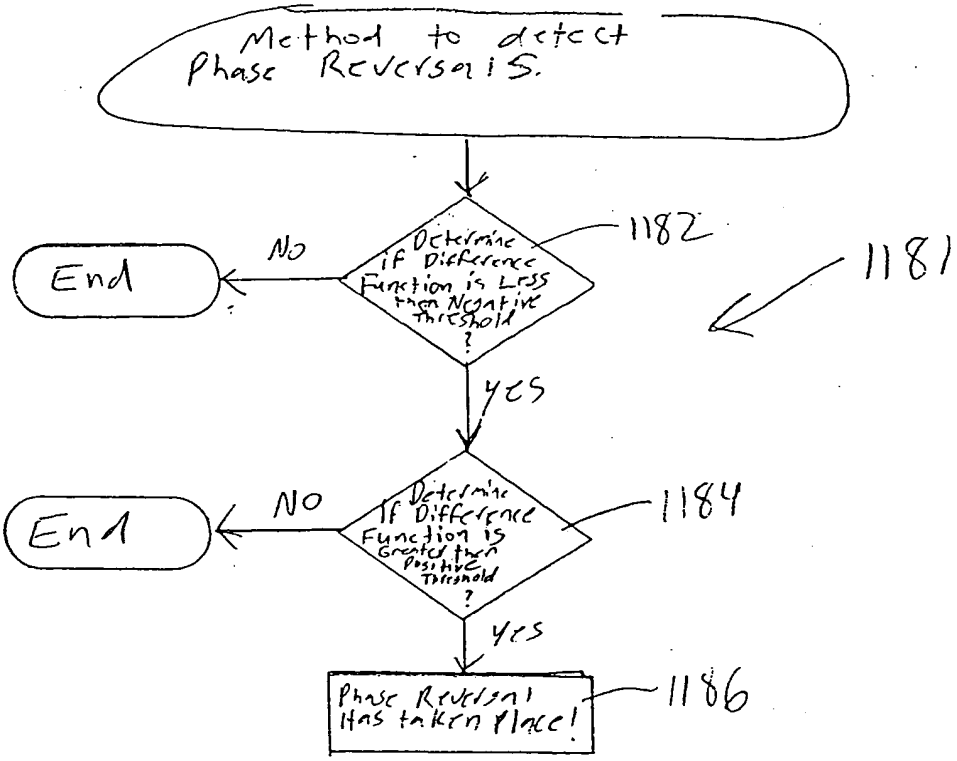

x[n]
y[n]
y1[n]
b10
b11
-a11
b12
-a12
d11[n]
d12[n]
b20
b21
-a21
b22
-a22
d21[n]
d22[n]
z-1

FIG. 11F


Method to detect Phase Reversals.
Determine if Difference Function is Less than Negative Threshold ?
1182
1181
No
End
yes
Determine if Difference Function is Greater than Positive Threshold ?
1184
NO
End
yes
Phase Reversal Has taken Place!
1186

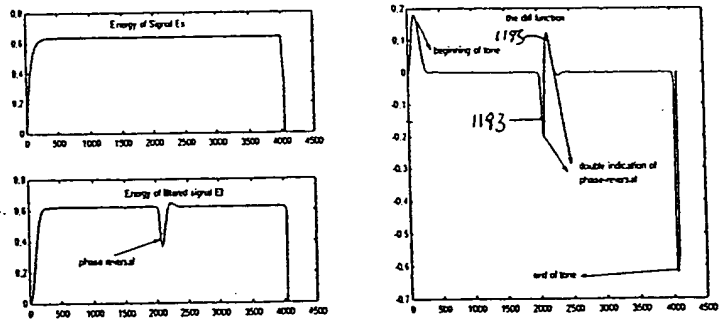

Energy of Signal Es
Energy of filtered signal Ef
phase reversal
the dif function
beginning of tone
1195
1193
double indication of phase-reversal
end of tone

FIG. 11G

Method for Fax V.21 Detection

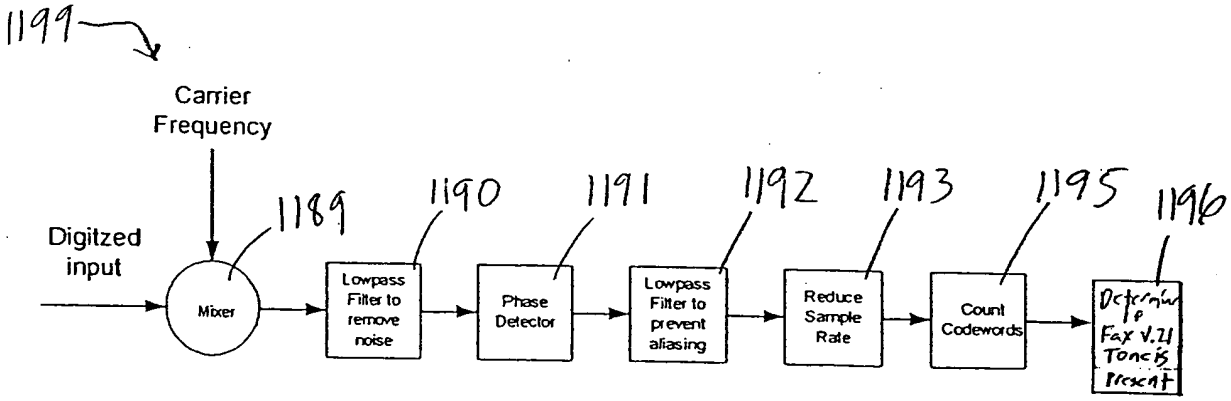


FIG. 11H

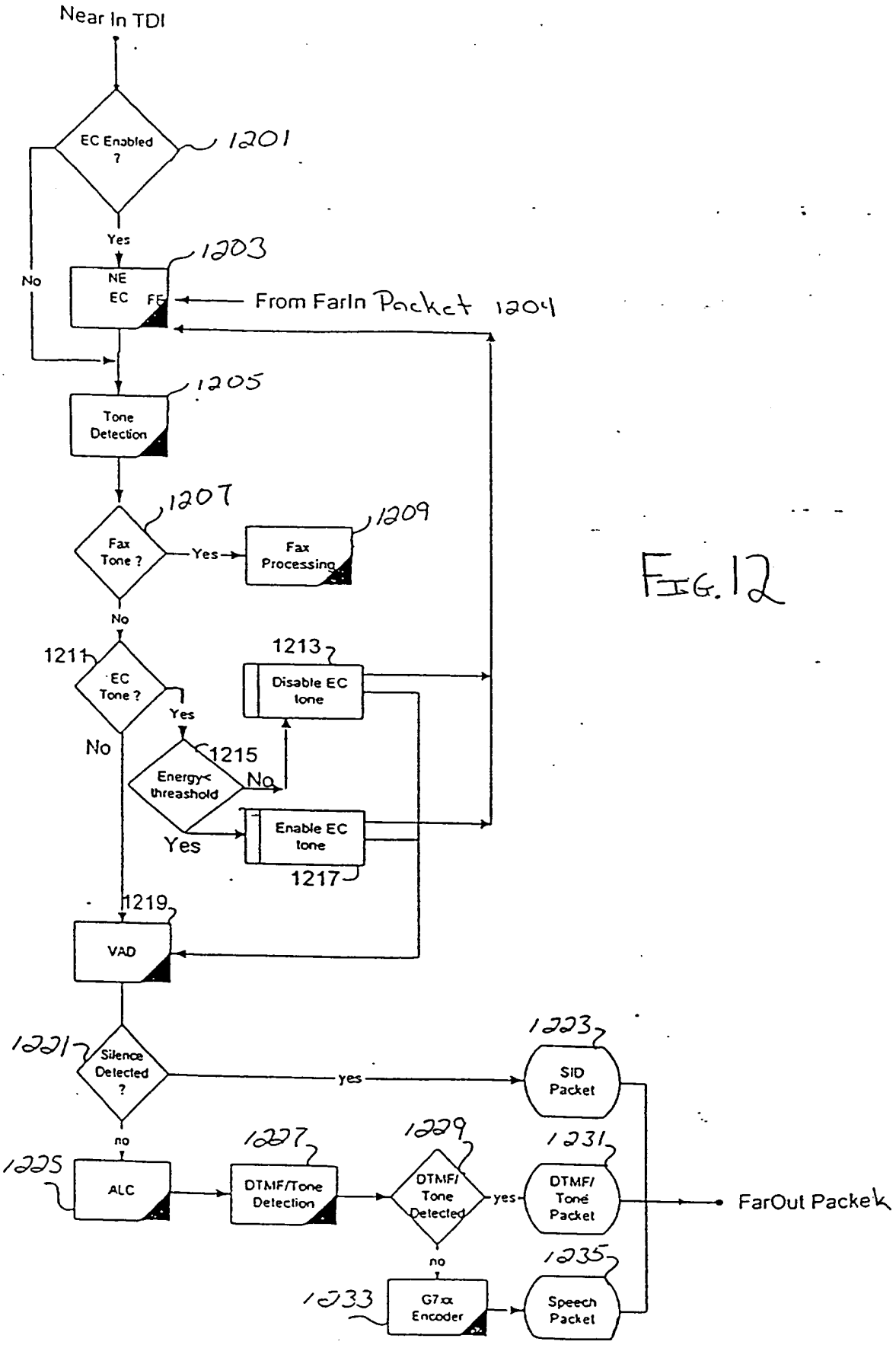

Near In TDI
EC Enabled ?
1201
Yes
No
NE EC FE
1203
From FarIn Packet 1204
Tone Detection
1205
Fax Tone ?
1207
Yes
Fax Processing
1209
No
EC Tone ?
1211
Yes
No
Disable EC tone
1213
Energy< threashold
1215
No
Yes
Enable EC tone
1217
VAD
1219
Silence Detected ?
1221
yes
SID Packet
1223
no
ALC
1225
DTMF/Tone Detection
1227
DTMF/ Tone Detected
1229
yes
DTMF/ Tone Packet
1231
FarOut Packet
no
G7xx Encoder
1233
Speech Packet
1235

FIG. 12

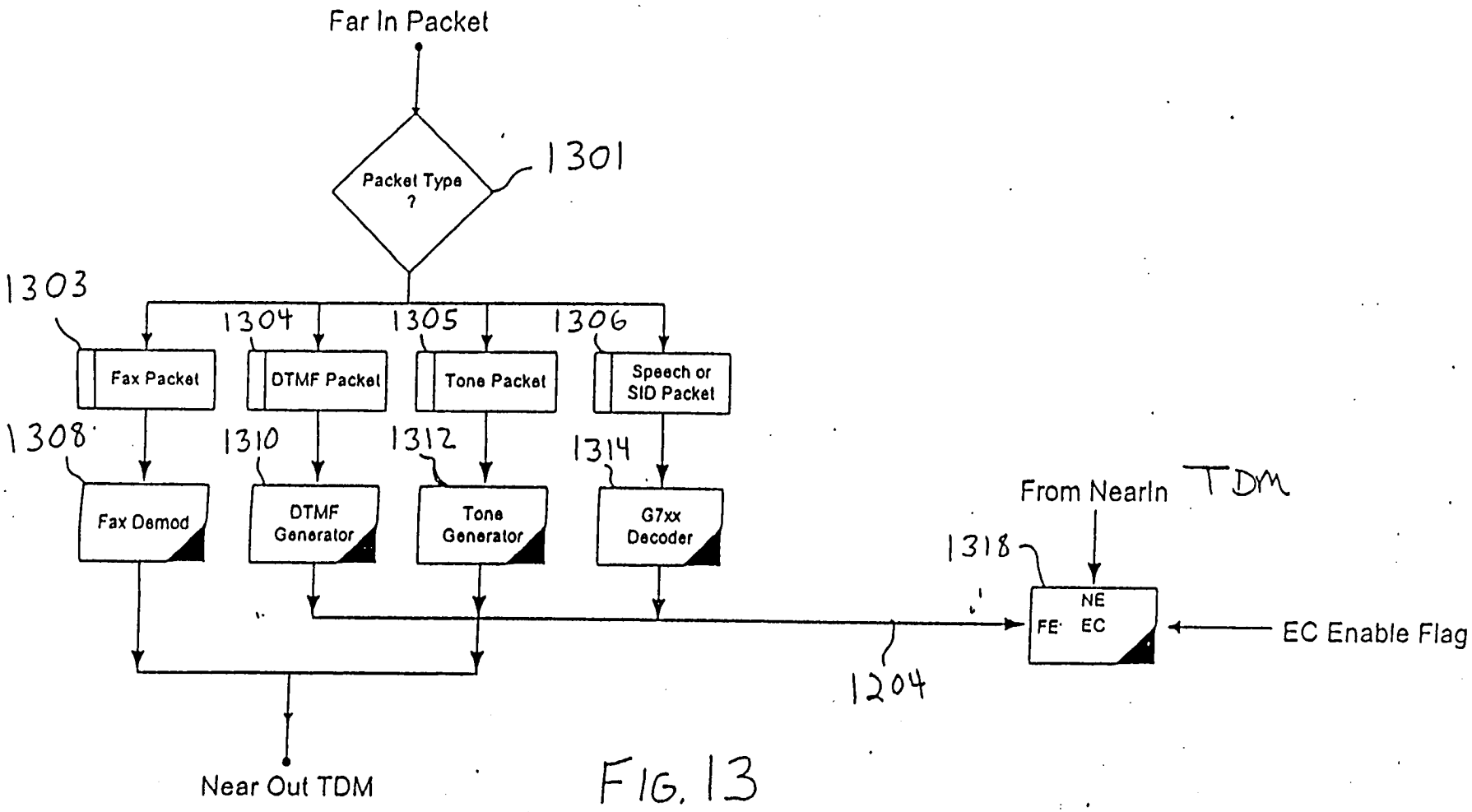

Far In Packet
Packet Type ?
1301
1303
1304
1305
1306
Fax Packet
DTMF Packet
Tone Packet
Speech or SID Packet
1308
1310
1312
1314
Fax Demod
DTMF Generator
Tone Generator
G7xx Decoder
From NearIn TDM
1318
NE
FE
EC
EC Enable Flag
1204
Near Out TDM

FIG. 13

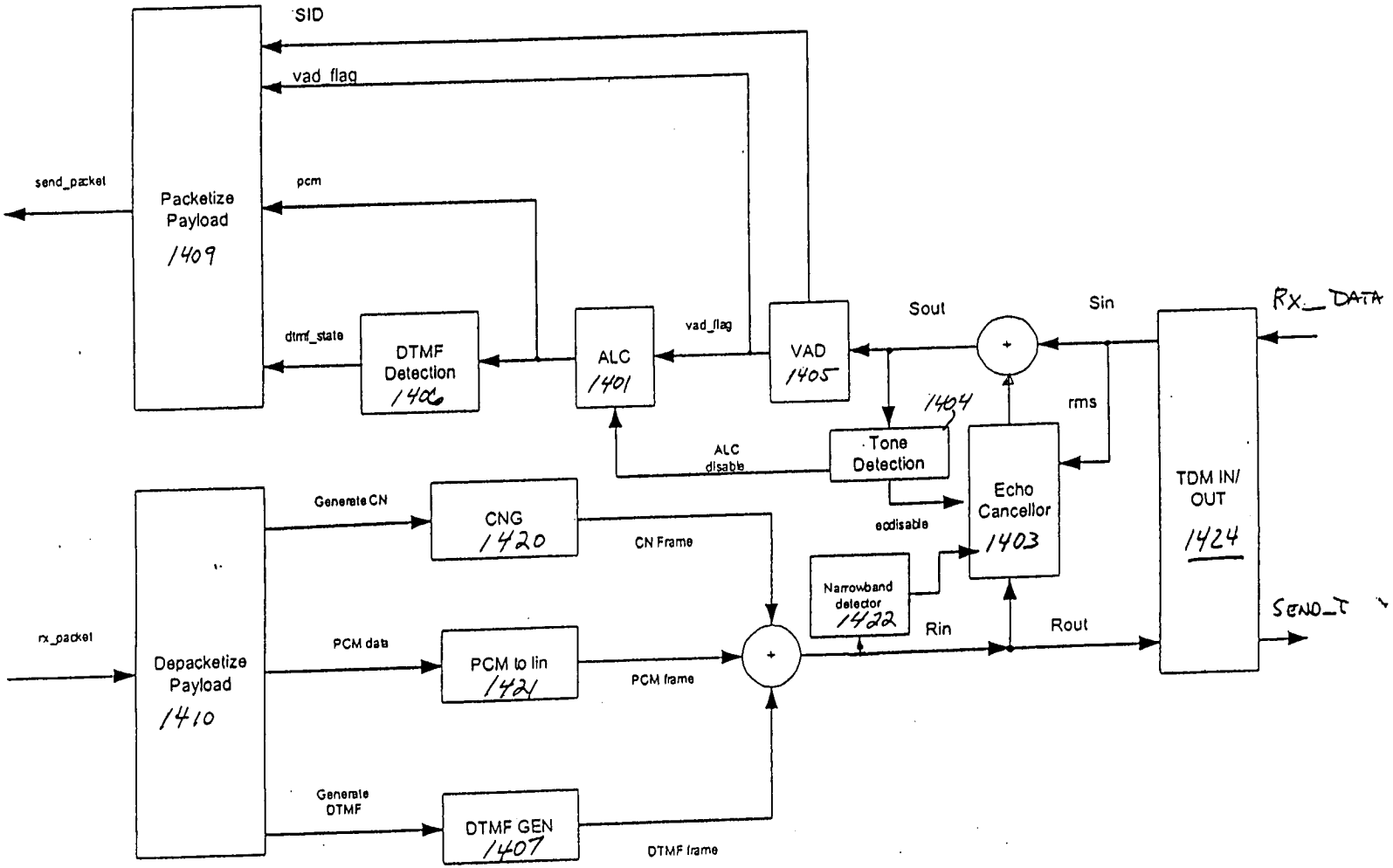

SID
vad_flag
send_packet
Packetize Payload
1409
pcm
dtmf_state
DTMF Detection
1406
ALC
1401
vad_flag
VAD
1405
Sout
Sin
RX_DATA
1404
rms
ALC disable
Tone Detection
Echo Cancellor
1403
TDM IN/ OUT
1424
edisable
Generate CN
CNG
1420
CN Frame
Narrowband detector
1422
Rin
Rout
SEND_T
rx_packet
Depacketize Payload
1410
PCM data
PCM to lin
1421
PCM frame
Generate DTMF
DTMF GEN
1407
DTMF frame

FIG. 14

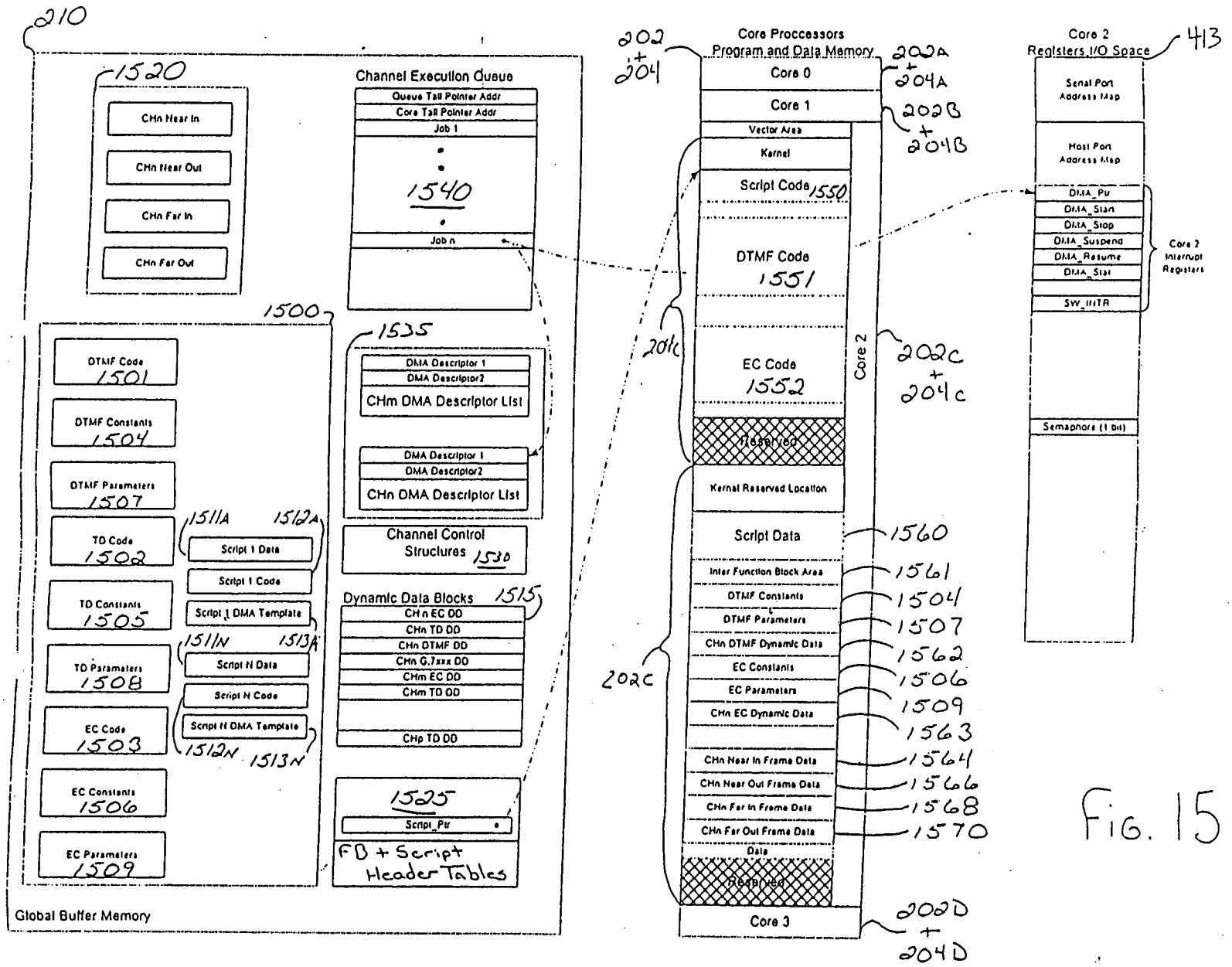

210
1520
CHn Near In
CHn Near Out
CHn Far In
CHn Far Out
Channel Execution Queue
Queue Tail Pointer Addr
Core Tail Pointer Addr
Job 1
1540
Job n
1500
DTMF Code 1501
DTMF Constants 1504
DTMF Parameters 1507
TD Code 1502
TD Constants 1505
TD Parameters 1508
EC Code 1503
EC Constants 1506
EC Parameters 1509
1511A
1512A
Script 1 Data
Script 1 Code
Script 1 DMA Template
1511N
1513A
Script N Data
Script N Code
Script N DMA Template
1512N
1513N
1535
DMA Descriptor 1
DMA Descriptor2
CHm DMA Descriptor List
DMA Descriptor 1
DMA Descriptor2
CHn DMA Descriptor List
Channel Control Structures 1530
Dynamic Data Blocks 1515
CH n EC DD
CHn TD DD
CHn DTMF DD
CHn G.7xxx DD
CHm EC DD
CHm TD DD
CHp TD DD
1525
Script_Ptr
FD + Script Header Tables
Global Buffer Memory
202 + 204
Core Processors Program and Data Memory
202A + 204A
Core 0
Core 1
202B + 204B
Vector Area
Kernel
Script Code 1550
DTMF Code 1551
EC Code 1552
Reserved
Kernal Reserved Location
Script Data
1560
Inter Function Block Area
1561
DTMF Constants
1504
DTMF Parameters
1507
CHn DTMF Dynamic Data
1562
EC Constants
1506
EC Parameters
1509
CHn EC Dynamic Data
1563
CHn Near In Frame Data
1564
CHn Near Out Frame Data
1566
CHn Far In Frame Data
1568
CHn Far Out Frame Data
1570
Data
Reserved
Core 3
202D + 204D
Core 2
202C + 204C
202C
Core 2 Registers,I/O Space
413
Serial Port Address Map
Host Port Address Map
DMA_Ptr
DMA_Start
DMA_Stop
DMA_Suspend
DMA_Resume
DMA_Stat
SW_INTR
Core 2 Interrupt Registers
Semaphore (1 bit)

Fig. 15

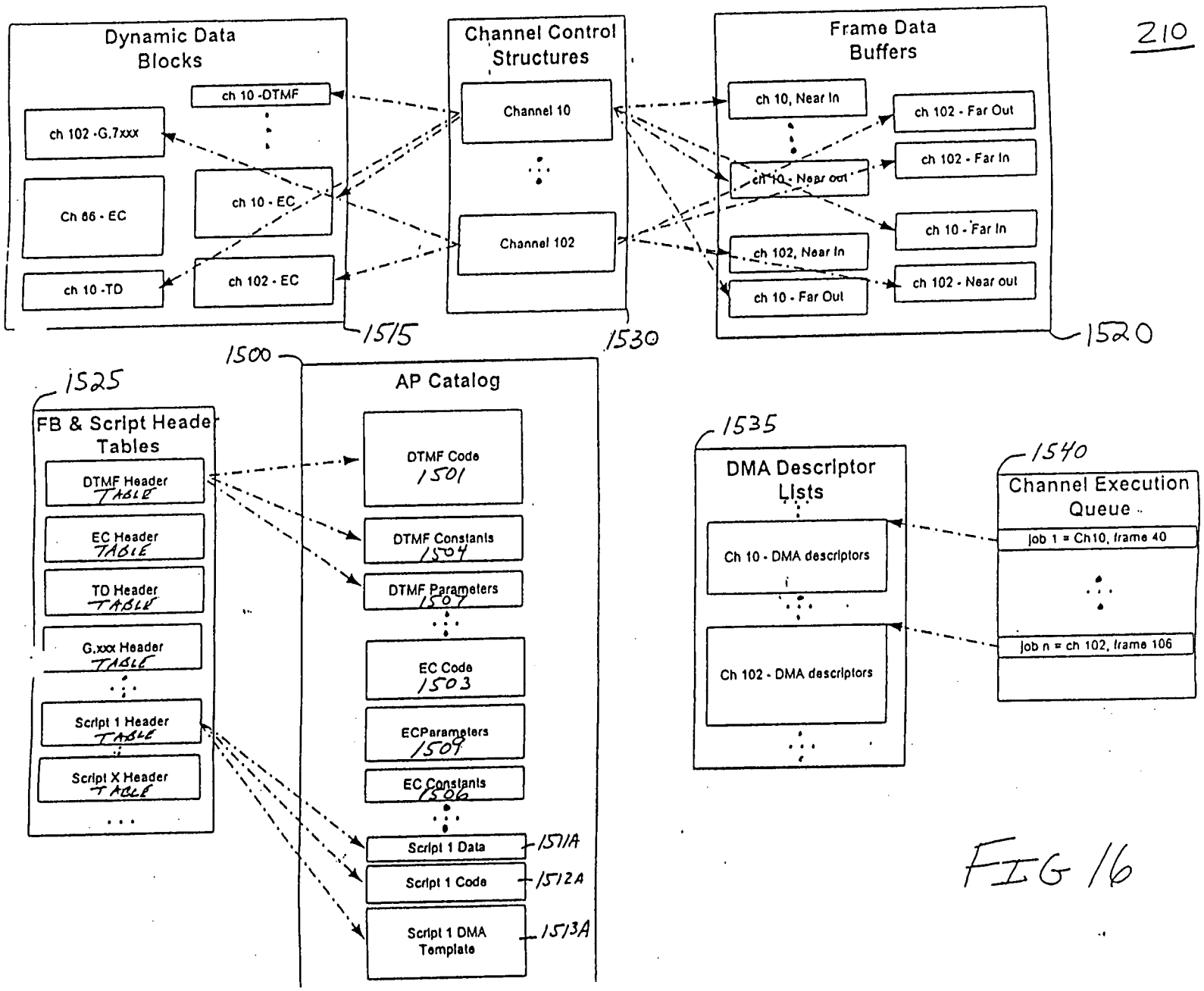

210
Dynamic Data Blocks
ch 10 -DTMF
ch 102 -G.7xxx
Ch 66 - EC
ch 10 - EC
ch 10 -TD
ch 102 - EC
1515
Channel Control Structures
Channel 10
Channel 102
1530
Frame Data Buffers
ch 10, Near In
ch 10 - Near out
ch 102, Near In
ch 10 - Far Out
ch 102 - Far Out
ch 102 - Far In
ch 10 - Far In
ch 102 - Near out
1520
1500
AP Catalog
1525
FB & Script Header Tables
DTMF Header TABLE
EC Header TABLE
TO Header TABLE
G.xxx Header TABLE
Script 1 Header TABLE
Script X Header TABLE
DTMF Code 1501
DTMF Constants 1504
DTMF Parameters 1507
EC Code 1503
ECParameters 1509
EC Constants 1506
Script 1 Data
Script 1 Code
Script 1 DMA Template
1511A
1512A
1513A
1535
DMA Descriptor Lists
Ch 10 - DMA descriptors
Ch 102 - DMA descriptors
1540
Channel Execution Queue
Job 1 = Ch10, frame 40
Job n = ch 102, frame 106

FIG 16

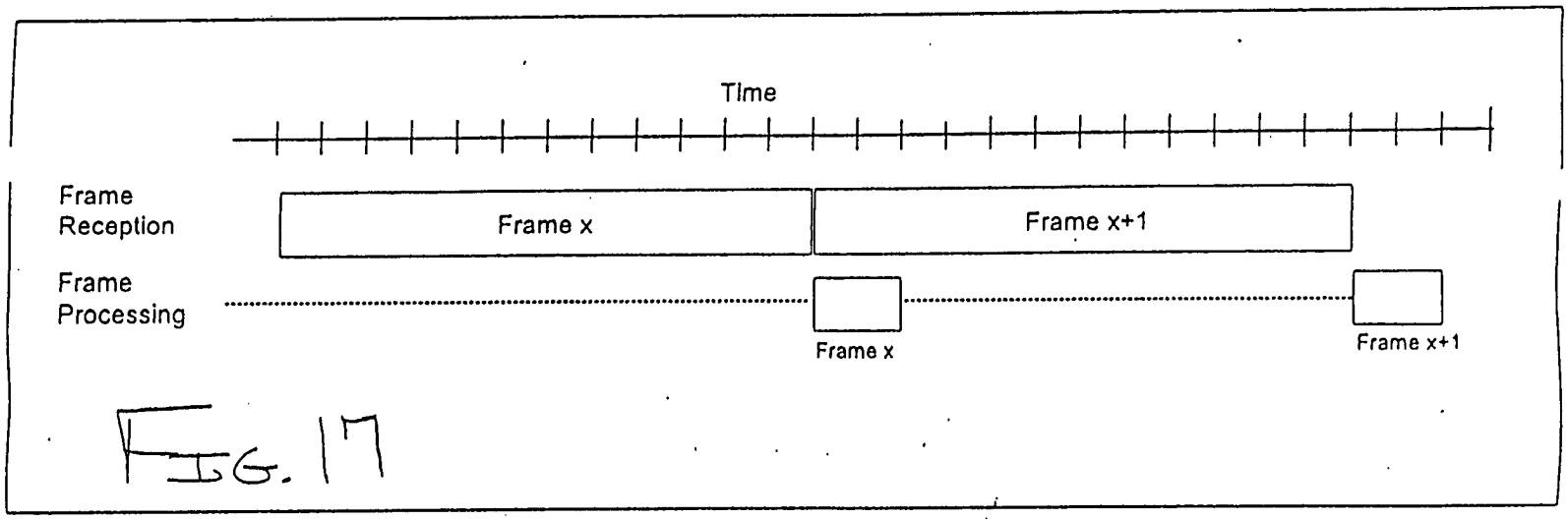
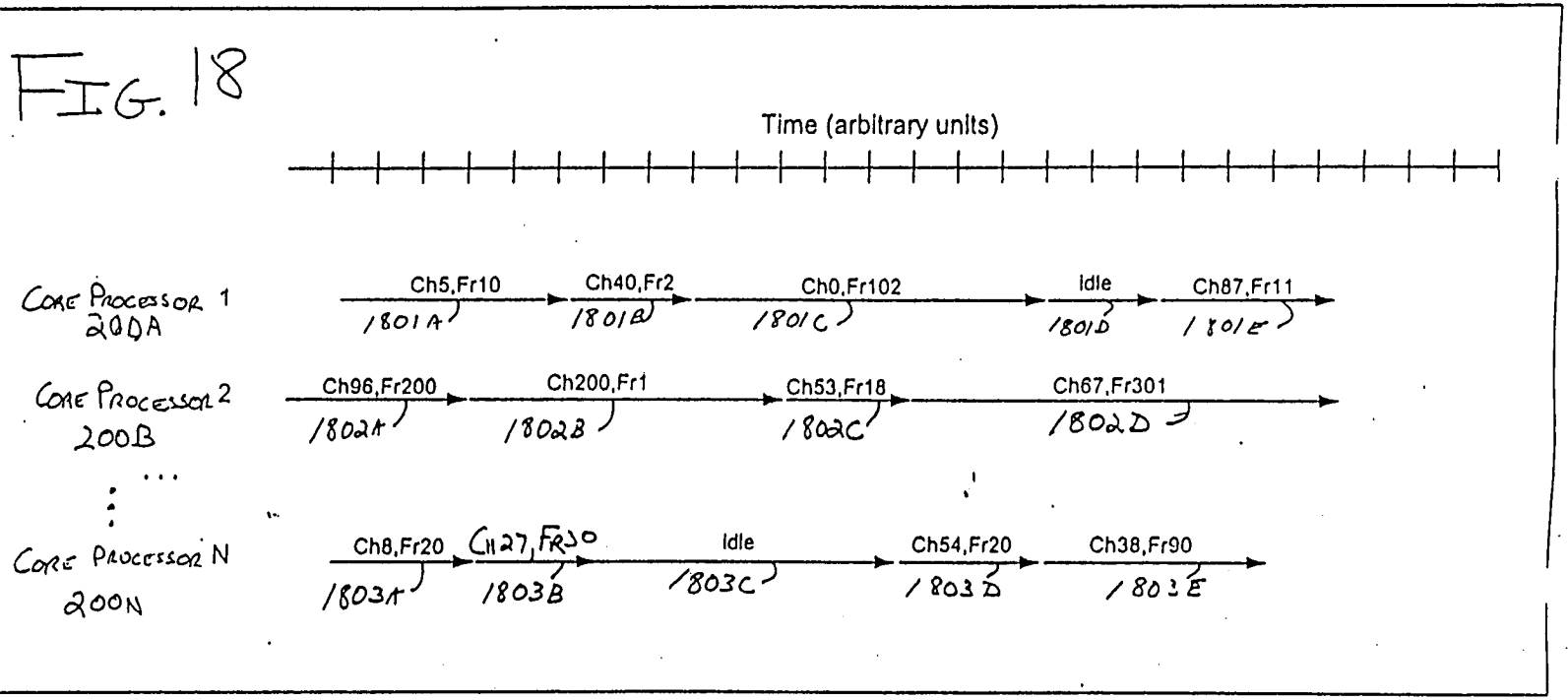

Time
Frame Reception
Frame x
Frame x+1
Frame Processing
Frame x
Frame x+1

FIG. 17

FIG. 18


Time (arbitrary units)
Core Processor 1
200A
Ch5,Fr10
1801A
Ch40,Fr2
1801B
Ch0,Fr102
1801C
Idle
1801D
Ch87,Fr11
1801E
Core Processor 2
200B
Ch96,Fr200
1802A
Ch200,Fr1
1802B
Ch53,Fr18
1802C
Ch67,Fr301
1802D
Core Processor N
200N
Ch8,Fr20
1803A
Ch27,Fr30
1803B
Idle
1803C
Ch54,Fr20
1803D
Ch38,Fr90
1803E